

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開 2001-125797

(P 2001-125797A)

(43) 公開日 平成13年5月11日 (2001. 5. 11)

(5) Int. Cl. 7	識別記号	F I	テマコード (参考)
G 0 6 F 9/46	3 4 0	G 0 6 F 9/46 3 4 0 B	5B042
G 0 5 B 19/18		11/30 3 0 5 D	5B098
G 0 6 F 11/30	3 0 5	G 0 5 B 19/18 X	5H269

審査請求 未請求 請求項の数 7

OL

(全 15 頁)

(21) 出願番号 特願平11-302347

(22) 出願日 平成11年10月25日 (1999. 10. 25)

(71) 出願人 000002369

セイコーエプソン株式会社

東京都新宿区西新宿2丁目4番1号

(72) 発明者 野上 大志

長野県諏訪市大和3丁目3番5号 セイコー

エプソン株式会社内

(74) 代理人 100061273

弁理士 佐々木 宗治 (外3名)

F ターム (参考) 5B042 JJ15

5B098 GA04 GB01 JJ08

5H269 AB01 AB33 BB12 EE11 PP03

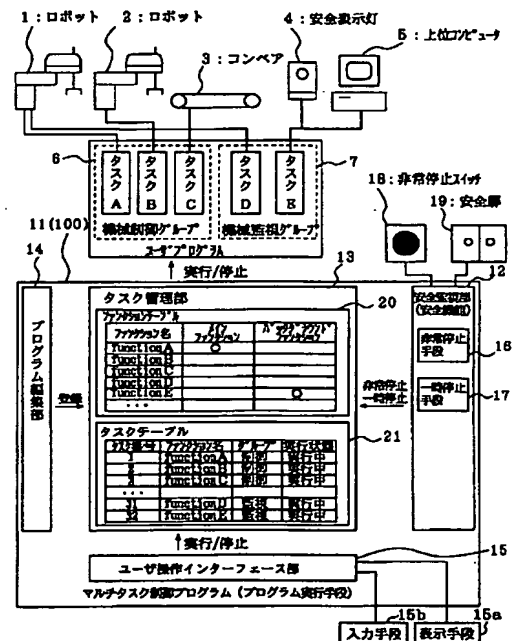
QB15 QE12 QE21 QE22

(54) 【発明の名称】 マルチタスクシステム及びそのプログラムを記録した記録媒体並びに加工装置

(57) 【要約】

【課題】 タスクをグループ化し、各グループ毎に安全機能の作用を異ならせることにより1つの制御装置で安全かつ簡単に機械の制御を行うと共に、常に各装置の監視が可能なマルチタスクシステム及びそのプログラムを記録した媒体並びに加工装置を提供する。

【解決手段】 複数のタスクを制御するマルチタスクシステム100であって、複数のタスクをマルチタスク処理するプログラム実行手段11と、複数のタスクのそれぞれに対して設定されたグループ化情報を保持するタスクテーブル21とを有し、プログラム実行手段11は、安全機能としての停止手段(非常停止手段16、一時停止手段17)を有し、停止手段は、停止指示に基づきタスクテーブル21からグループ化情報を取得して各タスクのグループをそれぞれ識別し、グループ毎にそれぞれ異なる制御を行うものである。



100: マルチタスクシステム

【特許請求の範囲】

【請求項 1】 複数のタスクを制御するマルチタスクシステムであって、
複数のタスクをマルチタスク処理するプログラム実行手段と、

複数のタスクのそれぞれに対して設定されたグループ化情報を保持するタスクテーブルとを有し、

前記プログラム実行手段は、安全機能としての停止手段を有し、

該停止手段は、停止指示に基づき前記タスクテーブルからグループ化情報を取得して各タスクのグループをそれぞれ識別し、グループ毎にそれぞれ異なる制御を行うことを特徴とするマルチタスクシステム。

【請求項 2】 前記停止手段は、前記タスクテーブルのグループ化情報に基づき識別した第 1 のグループのタスクに対しては、その実行を終了又は一時停止させ、第 2 のグループに属するタスクに対しては、その実行を継続させることを特徴とする請求項 1 記載のマルチタスクシステム。

【請求項 3】 前記第 1 のグループは、装置を制御するタスクからなる機械制御グループであり、前記第 2 のグループは、監視処理もしくは通信処理を行うタスクからなる機械監視グループであることを特徴とする請求項 2 記載のマルチタスクシステム。

【請求項 4】 複数のタスクを制御するマルチタスクシステムであって、

複数のタスクをマルチタスク処理するプログラム実行手段と、

複数のタスクのそれぞれが、

装置の監視処理もしくは外部装置との通信処理を行うタスクからなる機械監視グループ、及び装置の動作を制御するタスクからなる機械制御グループのいずれに属するか

のグループ情報と、
実行状態及び一時停止状態を含む各種状態のうち、いずれの状態にあるかの状態情報とを保持するタスクテーブルとを有し、

前記プログラム実行手段は、安全機能としての非常停止手段及び一時停止手段を有し、

前記非常停止手段は、非常停止指示に基づき前記タスクテーブルをアクセスして各タスクが属するグループを識別し、機械制御グループに属するタスクを終了させ前記タスクテーブルの内容を更新し、

前記一時停止手段は、一時停止指示に基づき前記タスクテーブルをアクセスして各タスクが属するグループを識別し、機械制御グループに属するタスクを一時停止させ前記タスクテーブルの内容を更新し、

前記非常停止手段及び一時停止手段は、機械監視グループに属するタスクに対して実行状態を継続させる機能を有することを特徴とするマルチタスクシステム。

【請求項 5】 前記タスクテーブルの内容を表示する表

示手段を設けたことを特徴とする請求項 1 乃至 4 のいずれかに記載のマルチタスクシステム。

【請求項 6】 請求項 1 乃至請求項 5 のいずれかに記載のマルチタスクシステムとして機能させるためのプログラムを記録した記録媒体。

【請求項 7】 請求項 1 乃至 5 のいずれかに記載のマルチタスクシステムと、前記機械制御グループによって制御されてなる第 1 の装置と、前記機械監視グループによって制御されてなる第 2 の装置と、を少なくとも有してなり、

前記マルチタスクシステムに構成される停止手段により、各装置が属するグループ毎に停止／継続の制御を行うことを特徴とする加工装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はマルチタスクシステム及びそのプログラムを記録した媒体並びに加工装置に関し、特に安全機能を有するマルチタスクシステム及びそのプログラムを記録した媒体並びに加工装置に関するものである。

【0002】

【従来の技術】従来、ロボット、周辺装置などの複数の装置を備えたシステムの制御に際して、複数のプログラム（タスク）を同時に実行するマルチタスク機能を備えたマルチタスクシステムが広く利用されている。特開平 7-129418 号公報には、各装置に対応するプログラム間に親子関係を設定し、親タスクが起動されると子タスクも自動的に実行させ、複数プログラムの起動を簡単にしたマルチタスク環境でのプログラム制御方式が開示されている。また、特開平 8-263127 号公報には通信機能を備えた外部装置に対する命令からなるタスクと、ロボット制御に関する命令からなるタスクとに分けてタスクを構成し、このように通信に係るタスクを制御に係るタスクと分けて構成することで、容易なプログラム記述を可能とした数値制御装置が開示されている。また、特開平 8-339211 号公報には、ロボットなどの各装置を制御するタスクとは別に、システム全体のエラー状態を監視する監視タスクを設け、当該監視タスクを各装置それぞれに割り付けたタスクと並列に処理し、どの装置でエラーが発生しても、その内容に従って瞬時に各タスクに適切な処理を指令することができる自動機における制御装置が開示されている。

【0003】

【発明が解決しようとする課題】ところで、通常、例えば工作機械や産業用ロボット等の装置をマルチタスクで制御する制御装置には、安全確保のため、例えば非常停止スイッチや一時停止スイッチ、安全柵等が接続され、非常停止スイッチが押下されたり、安全柵の扉が開かれたりした場合、可動部の駆動モータに供給される電力を遮断してロボットを停止させると共にタスクを停止させ

る非常停止手段や、ロボットとタスクとを一時停止させる一時停止手段といった安全機能が備わっている。

【0004】このような安全機能を、例えば特開平7-129418号公報に開示されている制御方式を適用した制御装置で使用した場合、まず親タスクが停止し、それにつられて子タスクも停止するようになっている。このように、従来技術では、安全機能が働いた場合、全てのタスクが停止する構成となっており、例えば上位コンピュータとの通信処理など、安全機能が働いたとしてもその実行が望まれる一部のタスクの実行を継続させるなどの配慮がなされていない。従って、非常停止時や一時停止時に機械の状態をオペレータや上位コンピュータに伝えることができず、十分な監視を行えないという問題点があった。

【0005】特開平8-263127号公報に開示された装置は、外部装置との通信に係るタスクを機械を制御するタスクと別に構成したものであるが、単に分けて構成しただけであり、上述したように安全機能が働いた場合であっても、通信に係るタスクのように必要部分のタスクに関しては実行を継続させ、その他の機械制御に係るタスクは通常通り停止させるようにする等、タスク毎に異なる制御を行うものではない。

【0006】なお、上述したような全タスクの停止を回避するには、特開平8-339211号公報に開示されたもののように、制御装置側の安全機能を使用するのではなく、ユーザによって安全機能をプログラミングする方法や、ロボット等の装置毎に制御装置を用いてシステムを構築する方法が有効であるが、前者の方法の場合、プログラムミスによる危険を回避できないばかりか、仮に監視タスクが停止してしまった場合、ロボットを動かしているタスクを停止させることもできず、非常停止スイッチを押してもロボットが停止しない等、非常に危険な状態が発生する恐れがあった。また、後者の方法の場合は、複数の制御装置が必要となるためコストがアップする上、制御装置間での同期処理が必要になり、プログラムが複雑化するという問題がある。

【0007】本発明は、このような点に鑑みなされたもので、各タスクが属するグループ毎に安全機能の作用を異ならせることにより1つの制御装置で安全かつ簡単に機械の制御を行うと共に、常に各装置の監視が可能なマルチタスクシステム及びそのプログラムを記録した媒体並びに加工装置を提供することを目的とする。

【0008】

【課題を解決するための手段】本発明に係るマルチタスクシステムは、複数のタスクを制御するマルチタスクシステムであって、複数のタスクをマルチタスク処理するプログラム実行手段と、複数のタスクのそれぞれに対して設定されたグループ化情報を保持するタスクテーブルとを有し、プログラム実行手段は、安全機能としての停止手段を有し、停止手段は、停止指示に基づきタスク

テーブルからグループ化情報を取得して各タスクのグループをそれぞれ識別し、グループ毎にそれぞれ異なる制御を行うものである。

【0009】また、本発明に係るマルチタスクシステムの停止手段は、タスクテーブルのグループ化情報に基づき識別した第1のグループのタスクに対しては、その実行を終了又は一時停止させ、第2のグループに属するタスクに対しては、その実行を継続させるものである。

【0010】さらに、本発明に係るマルチタスクシステムの第1のグループは、装置を制御するタスクからなる機械制御グループであり、第2のグループは、監視処理もしくは通信処理を行うタスクからなる機械監視グループである。なお、ここでの装置とは、例えば可動部を備えた機械である。

【0011】本発明に係るマルチタスクシステムは、複数のタスクを制御するマルチタスクシステムであって、複数のタスクをマルチタスク処理するプログラム実行手段と、複数のタスクのそれぞれが、装置の監視処理もしくは外部装置との通信処理を行うタスクからなる機械監視グループ、及び装置の動作を制御するタスクからなる機械制御グループのいずれに属するかのグループ情報と、実行状態及び一時停止状態を含む各種状態のうち、いずれの状態にあるかの状態情報とを保持するタスクテーブルとを有し、プログラム実行手段は、安全機能としての非常停止手段及び一時停止手段を有し、非常停止手段は、非常停止指示に基づき前記タスクテーブルをアクセスして各タスクが属するグループを識別し、機械制御グループに属するタスクを終了させ前記タスクテーブルの内容を更新し、一時停止手段は、一時停止指示に基づきタスクテーブルをアクセスして各タスクが属するグループを識別し、機械制御グループに属するタスクを一時停止させタスクテーブルの内容を更新し、非常停止手段及び一時停止手段は、機械監視グループに属するタスクに対して実行状態を継続させる機能を有するものである。

【0012】また、本発明に係るマルチタスクシステムは、タスクテーブルの内容を表示する表示手段を設けたものである。

【0013】本発明に係る記録媒体は、上記の各マルチタスクシステムのうち、いずれかのマルチタスクシステムとして機能させるためのプログラムを記録した記録媒体である。

【0014】本発明に係る加工装置は、上記の各マルチタスクシステムのうち、いずれかのマルチタスクシステムと、機械制御グループによって制御されてなる第1の装置と、機械監視グループによって制御されてなる第2の装置と、を少なくとも有してなり、マルチタスクシステムに構成される停止手段により、各装置が属するグループ毎に停止/継続の制御を行うものである。

【0015】

10

20

30

40

50

【発明の実施の形態】図1は本発明の一実施の形態のマルチタスクシステムを含むシステム構成の一例を示す図である。本実施の形態では、グループ化の例として、可動部を備えた機械の制御を行う機械制御グループと、機械の監視処理や上位コンピュータとの通信処理を行う機械監視グループとに分けた場合を例に説明する。

【0016】マルチタスクシステム100は、可動部を有する2台のロボット（ロボット1、ロボット2）及びコンベアー3を制御するように構成され、安全表示灯4及び上位コンピュータ5と接続されている。6は機械の制御を行うタスクから構成される機械制御グループで、ここではロボット1を制御するためのタスクA、ロボット2を制御するためのタスクB、コンベアー3を制御するためのタスクCを備えている。7は機械の監視処理及び外部装置である上位コンピュータ5との通信処理を行うタスクから構成される機械監視グループ7で、ここではロボット1の監視を行うタスクD、安全表示灯4のON・OFF及び上位コンピュータ5との通信を行うタスクEを備えている。これらの各タスクはユーザにより作成されたユーザプログラムで、ユーザによりグループ分けが行われる。なお、機械制御グループ6のタスクには機械制御を行う命令の他、監視命令や通信命令が含まれていても良いが、機械監視グループ7のタスクには機械の制御命令は含まないように構成される。

【0017】11はマルチタスク処理が可能なCPU（図示せず）と安全機能を有するマルチタスク制御プログラム（以下、アプリケーションという）により実現されるプログラム実行手段で、図に示すように、安全監視部12、タスク管理部13、プログラム編集部14、ユーザ操作インタフェース部15を備えた構成となっている。

【0018】安全監視部（安全機能）12は、各タスクでプログラムされた命令とは無関係にタスクや機械（ロボット1、ロボット2及びコンベアー3）を停止する部分で、非常停止手段16と一時停止手段17を備えている。非常停止手段16は、例えば非常停止スイッチ18が押下されたり、安全柵19が開かれる等した場合、機械の駆動モータ（図示せず）への電力供給を遮断して機械を停止させると共に機械制御グループ6に属するタスクを停止（以下では、一時停止との混同を避けるために終了と記す）させる。一時停止手段17は、例えば一時停止ボタンが押された場合に、機械及び機械制御グループ6に属するタスクを一時的に停止させる。なお、非常停止手段16、一時停止手段17とも、機械監視グループ7に属するタスクに対しては、その実行を継続させるようになっている。

【0019】タスク管理部13は、ファンクションテーブル20、タスクテーブル21に基づいてタスクの管理を行う部分で、ファンクションテーブル20には、ユーザによりプログラムされたファンクション（タスク）が

全て登録されると共に、後に詳述するメインファンクション、バックグラウンドファンクションの指定情報が格納される。タスクテーブル21には、タスク番号、ファンクション名、グループ化情報（何れのグループに属するか）の識別情報、タスクの状態が格納される。

【0020】プログラム編集部14は、ユーザプログラムの編集を行う部分である。ユーザ操作インタフェース部15は、表示手段15a、入力手段15bを用いた操作入力を実現とするGUI環境を提供する部分である。

【0021】以上のようにして構成されたプログラム実行手段11は、ファンクションテーブル20に登録された複数のファンクション（タスク）をマルチタスク処理で実行する。また、プログラム実行手段11は、各タスクのグループ違反等を検出するエラー検出機能を備えており、機械監視グループ7に属するタスクに機械の動作命令が含まれていた場合に、実行を中断しエラーを表示する。なお、この実行時のエラー検出に先だて、以下に詳述するタスク登録時にもエラー検出を行っており、このようにタスク登録時及び実行時の2段階でエラーチェックを行うことにより、機械監視グループ7に属するタスクにおいて、GO、JUMPといったロボット動作命令は実行できないように構成されている。

【0022】図2はアプリケーションによる表示画面の一例を示す図で、31はエディタ画面、32はオペレータ操作画面を示している。以下、タスクを機械制御グループ6と機械監視グループ7とに分類する手順について、（1）タスクの作成、（2）タスクの登録の順に説明する。

【0023】（1）タスク（ファンクション）の作成
タスクはファンクションを実行することによって生成されるもので、このファンクションはエディタ画面31で作成されたユーザプログラムである。

【0024】図3はユーザプログラムの一例を示す図で、図には、AからEの5つのファンクションを例示している。なお、このファンクションA～Eは図1のタスクA～Eにそれぞれ対応しており、ファンクションAは他のファンクションの実行を指示する命令（XQT）によりファンクションB及びCを起動すると共に、ロボット1に対する動作指示を行うプログラム、ファンクションB及びファンクションCは、それぞれロボット2及びコンベアー3に対する動作指示を行うプログラム、ファンクションDはロボット1の監視を行うプログラムである。ファンクションEはファンクションDを起動すると共に、非常停止時でも安全表示灯4のON、OFF命令の実行を行うプログラムである。このON、OFF命令はデフォルトで使用了場合、非常停止時には実行されないよう構成されているが、FORCEDキーワードにより、非常停止時でも強制的に実行させることができるようになっている。このようなFORCEDキーワードにより、ファンクションEに示した安全表示灯4のよう

に、非常時停止時であってもON、OFF命令の実行が望まれる装置に対しては実行可能となっている。

【0025】(2)タスク(ファンクション)の登録エディタ画面31で作成されたファンクションは、プログラム編集部14によって自動的にファンクションテーブル20に登録され、図2のファンクションリスト画面33に示すように登録されたファンクション名が表示される。そして、オペレータはオペレータ操作画面32上でボタン34、35を用いた所定の操作を行うことにより、メインファンクション、バックグラウンドファンクションを1つずつ指定する。メインファンクションに指定されたファンクションは、オペレータにより実行ボタン36が押下されたときに機械制御グループ6のタスクとして起動し、バックグラウンドファンクションに指定されたファンクションは、アプリケーションが立ち上がり、初期化が終了した時点で自動的に機械監視グループ7のタスクとして起動する。

【0026】なお、ファンクションAのように、他のタスクを起動する命令が含まれている場合は、親タスクに対するグループ指定が子タスクにも反映され、子タスクとして起動されたタスクB及びタスクCは、親タスクAと同じグループとして分類される。このように、メインファンクション、バックグラウンドファンクションの指定を行うことで、タスクを機械監視グループ7又は機械制御グループ6に分類する。

【0027】ここでは、ファンクションAをメインファンクションに指定し、ファンクションEをバックグラウンドファンクションに指定すると、ファンクションAは機械制御グループ6のタスクとなり、ファンクションAを親タスクとする子タスクB及び子タスクCも機械制御グループ6のタスクとなる。また、ファンクションEは機械監視グループ7のタスクとなり、ファンクションEを親タスクとする子タスクDも機械監視グループ7のタスクとなる。

【0028】図4はこの指定前後のファンクションテーブルを示す図で、指定されたそれぞれのファンクションに「○」印が書き込まれ、ファンクションテーブル20が更新される。

【0029】以上に説明したタスク登録時、プログラム実行手段11によりエラーチェックが行われ、バックグラウンドファンクションに指定されたファンクション内にロボット動作命令が含まれていないかをチェックし、含まれている場合はエラーを表示する。本例の場合、ファンクションAがバックグラウンドファンクションに指定された場合、エラーを表示する。なお、ここでのエラー検出の対象となるのは、指定されたファンクションのみであり、当該ファンクションを親タスクとする子タスクは対象外である。というのは、子タスクとして起動されるファンクションは、親タスクから起動されて始めてグループ決定がなされるためであり、グループ決定が行

われていない状態でのグループ違反の検出は困難であるからである。そこで、上述したように、登録時と実行時の両方でエラー検出を行うことで確実にグループ違反エラーを発見できるようになっている。

【0030】図5は以上のようにして作成登録されたタスクの起動手順を示すフローチャートである。なお、ここではタスクは最大32個実行できるものとし、機械制御グループ6のタスクはタスク番号1から昇順に割り当てられ、機械監視グループ7のタスクはタスク番号32から降順に割り当てられるように構成されているものとする。アプリケーションが起動されると、プログラム実行手段11はアプリケーションの初期化を行い(S1)、続いてファンクションテーブル20を検索してバックグラウンドファンクションが指定されているか否かを判断する(S2)。ここでは、ファンクションEが指定されているため、当該ファンクションEを機械監視グループ7に属するタスク番号32のタスク(タスクE)として起動し(S3)、オペレータ操作画面32を表示する(S4)。このとき、起動したタスクEはファンクションDを機械監視グループ7に属するタスク番号31のタスク(タスクD)として起動する。そして、ロボット動作を開始するための実行ボタン36が押されると、ファンクションテーブル20を検索してメインファンクションに指定されたファンクションAを機械制御グループ6に属するタスク番号1のタスク(タスクA)として起動する(S5)。このタスクAによってファンクションB、ファンクションCが機械制御グループ6に属するタスク番号2、タスク番号3のタスクとしてそれぞれ順に起動する。

【0031】このように、バックグラウンドファンクションに指定されたファンクション、すなわち機械監視グループ7に属するタスクをアプリケーション立ち上げ時に自動的に起動させることで、マルチタスクシステム100の起動時から監視処理及び通信処理を開始できるように構成されている。

【0032】図6は各状態におけるタスクテーブルを示す図で、図において①はアプリケーション初期化後、②はバックグラウンドファンクション起動後、③はファンクションEからファンクションDが起動された後、④はメインファンクション起動直後、⑤はファンクションAからファンクションBが起動された後、⑥はファンクションAから更にファンクションCが起動された後のタスクテーブル21を示している。このように、各状態におけるタスク情報によってタスクテーブル21が順次更新される。

【0033】図7はタスクテーブルの内容を表示したオペレータ操作画面の一例を示す図で、特に、図6の③の状態、図6の⑥の状態における表示例を示している。③の状態においては、起動中のタスクD、Eがタスクリスト画面38に表示され、それぞれ該当するタスク番号の

タスクマーカ37がチェックされる。⑥の状態においては、更にタスクA、B、Cが追加表示される。なお、図示されていないが、タスクリスト画面38の文字色は各タスクが属するグループに応じて異ならせて表示され、また、タスクマーカ37の表示色は各タスクの状態（実行中と一時停止中（後述する））によって異ならせて表示され、ユーザはオペレータ操作画面32によってタスクテーブル21の内容が確認できるようになっている。

【0034】次に、非常停止時の処理の流れについて図面を参照しながら説明する。図8は非常停止時の処理の流れを示すフローチャートである。オペレータにより非常停止スイッチ18が押下されるなどの非常停止指示があると、非常停止手段16は、ロボット1、ロボット2、コンベアー3をハード的に停止する（S11）。そして、タスクテーブルにタスクがあるか否かを1番目からチェックし（S12、S13）、ある場合には、続いて当該タスクが機械制御グループ6であるか否かを判断し（S14）、機械制御グループ6の場合にはタスクを終了させ（S15）、機械監視グループ7の場合には実行を継続させてステップS13に戻り、次のデータ（S16）で同様の処理を行う。これを繰り返し行い、タスクが存在しないN番目までくると、非常停止時の処理を終了する。

【0035】図9は非常停止前後のタスクテーブルを示す図で、非常停止前に存在していた機械制御グループ6のタスク（タスク番号1、2）が、非常停止後、タスクテーブル21から消去される。

【0036】次に、一時停止時の処理の流れについて図面を参照しながら説明する。図10は一時停止時の処理の流れを示すフローチャートである。オペレータにより一時停止ボタン39が押下されるなどの一時停止指示があると、一時停止手段17はロボット1、ロボット2、コンベアー3をハード的に停止する（S21）。そして、タスクテーブル21にタスクがあるか否かを1番目からチェックし（S22、S23）、ある場合には、当該タスクが機械制御グループ6であるか否かを判断し（S23）、機械制御グループ6の場合にはタスクを一時停止し（S24）、機械監視グループ7の場合には実行を継続させてステップS23に戻り、次のデータ（S26）で同様の処理を行う。これを繰り返し行い、タスクが存在しないN番目までくると、機械監視グループ7のタスクを一時停止にする処理は完了し、継続実行ボタン40の押下待ち状態となる。そして、継続実行ボタン40が押下されると（S27）、タスクテーブル21を検索して一時停止中のタスクを継続実行（再開）させ、一時停止時の処理を終了する。

【0037】図11は一時停止前、一時停止中、再開後のタスクテーブルを示す図で、図に示すようにタスク番号1、タスク番号2及びタスク番号3の機械制御グループ6のタスクの状態が実行状態から一時停止状態へと変

化し、そして、再開すると、一時停止状態から実行状態へと変化することが示されている。

【0038】このように、本実施の形態によれば、各タスクをグループ化し、グループ毎に異なる制御を行うようにしたので、各グループで必要とされる安全のレベルに応じた制御が可能となる。具体的には、上述したように機械制御グループ6のように可動部を備えた機械の制御を行うグループに対しては、当該グループのタスクを終了又は一時停止させ、他の機械監視グループ7に対しては実行を継続させるようにした。これにより、機械制御グループ6で制御される機械に対する安全を確実に確保しつつ、機械監視グループ7での監視及び通信処理を継続することが可能となった。

【0039】また、このようなタスクの終了や一時停止等の制御は、プログラム実行手段11の安全機能により行わせるようにしているので、ユーザが安全機能をプログラムする方法に比べ、プログラムミスによる危険を回避でき、安全なシステムを構成することができる。また、プログラム実行手段11が1つであるため、プログラミングが簡単でシンプルなシステムを構成することができる。

【0040】また、非常停止又は一時停止が指示された場合であっても、全てのタスクを終了させることなく、機械監視グループ7に属するタスクは実行を継続させるようにしたので、安全機能が働いてもロボット1、ロボット2、コンベアー3の状態を監視することができる。また、機械監視グループ7のタスクのプログラム次第で、非常停止中又は一時停止中に例えば安全表示灯4を点滅させる等ユーザがその表示形式をカスタマイズすることが可能となる。

【0041】また、機械監視グループ7に属するタスクはアプリケーションの初期化終了と同時に起動するため、マルチタスクシステム100立ち上げ時に非常停止状態にあった場合、上記と同様に安全表示灯4を点滅させたり、また、警告メッセージを表示したりすることが可能となる。

【0042】また、常に上位コンピュータ5との通信が可能となっているので、上位コンピュータ5でのリモート管理が可能となり、また、例えば休日・夜間無人運転時に何らかのエラーが発生した場合、エラーが発生したことを電子メールで送信することなども可能となる。

【0043】なお、本実施の形態では、各タスクを機械制御グループ6又は機械監視グループ7の2つに分類する場合を例に説明したが、これに限られたものではなく、例えば機械制御グループ1、機械制御グループ11、機械監視グループ等、3つにグループ化するようにしても良い。また、可動部を備えた機械全ての制御を1グループ（機械制御グループ6）にまとめた場合を例に説明したが、例えば機械制御グループ1にロボット1及びロボット2、機械制御グループ11にコンベアーを制御させる

ようにしてもよく、このようにグループ化した上で各グループ毎に非常停止スイッチを設けた場合、対応する非常停止スイッチが押下されたグループの機械のみを停止させることが可能となり、これにより複数の機械制御グループを独立して制御することが可能となる。

【0044】

【発明の効果】以上に説明したように本発明によれば、安全機能としての停止手段により各タスクが属するグループ毎にそれぞれ異なる制御を行うようにしたので、各グループで必要とされる安全のレベルに応じた制御が可能となる。

【0045】また、停止手段は、停止指示に基づき、全グループではなくあるグループのタスクに限り終了又は一時停止させるようにしたので、当該グループに、例えば可動部を備えた機械を制御するタスクを分類するようになれば、安全を確保しつつ所望の監視を継続させることが可能となる。また、このようなタスクの終了や一時停止の制御は、プログラム実行手段の停止手段が行うため、従来のユーザが安全機能をプログラムする方法に比べ、プログラムミスによる危険を回避でき、安全なシステムを構成することができる。

【0046】また、プログラム実行手段の安全機能により、機械の制御を行う機械制御グループのタスクは停止させ、監視処理及び通信処理を行う機械監視グループに対しては実行を継続させるようにしたので、1つのプログラム実行手段で機械の制御を安全かつ簡単に行うことができると共に、常に各装置の監視を行うことが可能となる。

【図面の簡単な説明】

【図1】本発明の一実施の形態のマルチタスクシステムを含むシステム構成の一例を示す図である。

【図2】アプリケーションによる表示画面の一例を示す

図である。

【図3】ユーザプログラムの一例を示す図である。

【図4】メインファンクション及びバックグラウンドファンクションの指定前後のファンクションテーブルを示す図である。

【図5】タスクの起動手順を示すフローチャートである。

【図6】各状態におけるタスクテーブルを示す図である。

【図7】タスクテーブルの内容を表示したオペレータ操作画面の一例を示す図である。

【図8】非常停止時の処理の流れを示すフローチャートである。

【図9】非常停止前後のタスクテーブルを示す図である。

【図10】一時停止時の処理の流れを示すフローチャートである。

【図11】一時停止前、一時停止中、再開後のタスクテーブルを示す図である。

【符号の説明】

1、2 ロボット

3 コンベア

5 上位コンピュータ（外部装置）

6 機械制御グループ

7 機械監視グループ

11 プログラム実行手段

16 非常停止手段

17 一時停止手段

15a 表示手段

21 タスクテーブル

100 マルチタスクシステム

【図9】

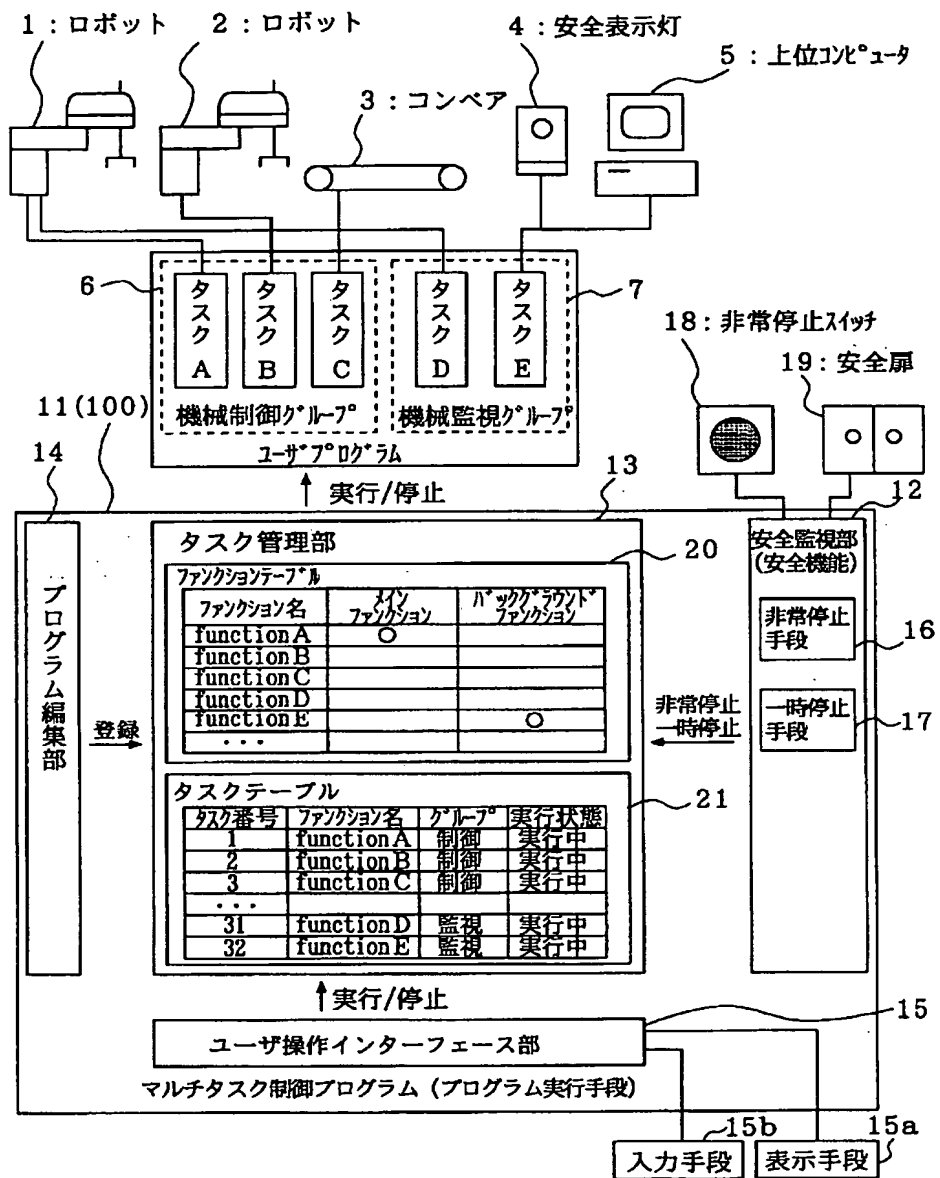
①非常停止前

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
2	Function B	制御	実行中
3	Function C	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

②非常停止後

タスク番号	ファンクション名	グループ	実行状態
31	Function D	監視	実行中
32	Function E	監視	実行中

【図1】



100: マルチタスクシステム

Figure 1 is a schematic diagram of a control system interface. The interface is divided into several sections. At the top, there's a header area with labels 31, 33, 37, 38, and 39. Below this, there's a main display area with labels 32, 34, 35, 36, and 40. The display area contains a 'FUNCTION A' section with a list of functions (1-16) and a 'FUNCTION B' section with a list of functions (1-16). The 'FUNCTION A' section also includes a 'SUB 1' and 'SUB 2' section. The 'FUNCTION B' section includes a 'SUB 1' and 'SUB 2' section. The interface also features a 'MENU' button (32), a 'START' button (34), a 'STOP' button (35), a 'RESET' button (36), and a 'PAUSE' button (40). The interface is designed to allow a user to select a function and execute it.

【図3】

```

FUNCTION A
  XQT B; REM ファンクションBを制御タスクとして起動する
  XQT C; REM ファンクションCを制御タスクとして起動する

  EXTERN GLOBAL n AS INTEGER
  n = 0; REM ロボット1の動作回数を保存する変数

  SELRB 1; REM ロボット1を選択
  MOTOR ON
  DO
    GO .動作位置1
    GO .動作位置2
    n = n+1; REM 動作回数をインクリメント
  LOOP
FEND

FUNCTION B
  SELRB 2; REM ロボット2を選択
  MOTOR ON
  DO
    GO .動作位置3
    GO .動作位置4
  LOOP
FEND

FUNCTION C
  DO
    ON コンベヤ
    SLEEP 3
    OFF コンベヤ
    SLEEP 1
  LOOP
FEND

FUNCTION D
  GLOBAL n AS INTEGER
  DO
    PRINT "ロボット1の動作回数 ",n
    SLEEP 10
  LOOP
FEND

FUNCTION E
  XQT D; REM ファンクションDを監視タスクとして起動する
  REM #21ポートへPRINTすることにより、上位コンピュータへRS-232C回線で、状態を出力
  DO
    IF (STAT(0) AND &H100000) <> 0 THEN
      PRINT #21, "非常停止入力により停止中"
      OFF 安全表示灯 FORCED
    ELSE
      PRINT #21, "アプリケーション正常動作中"
      ON 安全表示灯 FORCED
    ENDIF
    SLEEP 5
  LOOP
FEND

```

【図4】

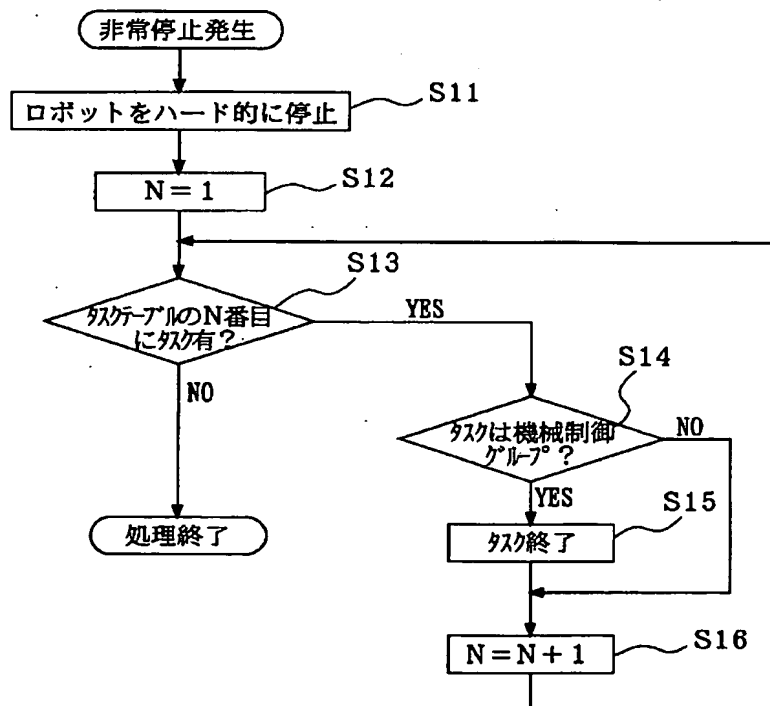
①初期状態

ファンクション名	メイン ファンクション	バックグラウンド ファンクション
function A		
function B		
function C		
function D		
function E		

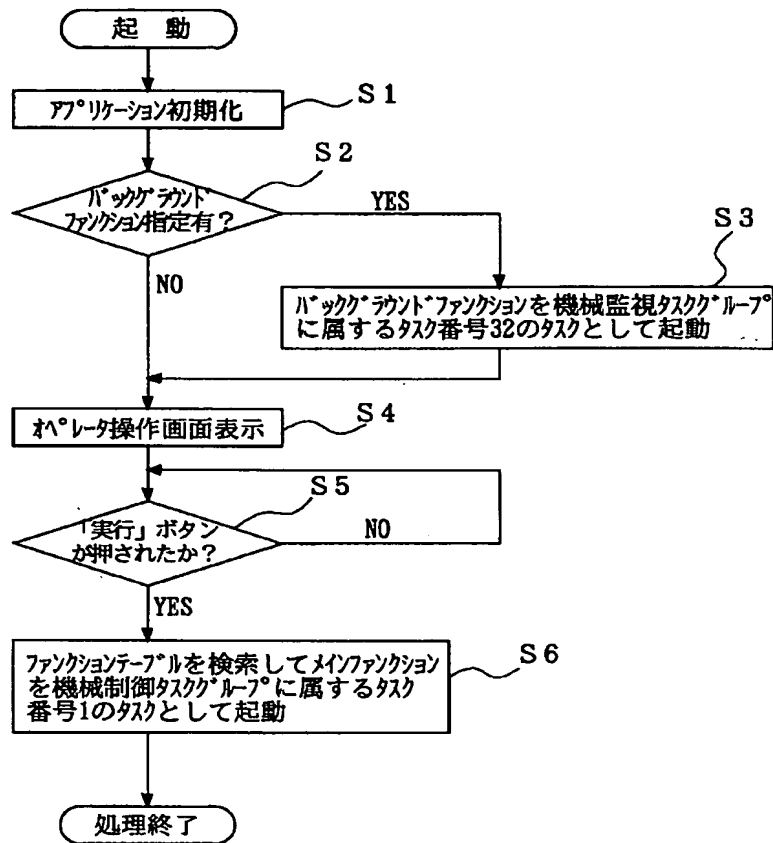
②ファンクションAをメインにファンクションBをバックに指定後

ファンクション名	メイン ファンクション	バックグラウンド ファンクション
function A	○	
function B		
function C		
function D		
function E		○

【図8】



【図 5】



【図 6】

①アプリケーション初期化後

タスク番号	ファンクション名	グループ	実行状態

②バックグラウンド・ファンクション起動直後

タスク番号	ファンクション名	グループ	実行状態
32	Function E	監視	実行中

③ファンクション E からファンクション D が起動された後

タスク番号	ファンクション名	グループ	実行状態
31	Function D	監視	実行中
32	Function E	監視	実行中

④メインファンクション起動直後

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

⑤ファンクション A からファンクション B が起動された後

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
2	Function B	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

⑥ファンクション A からファンクション C が起動された後

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
2	Function B	制御	実行中
3	Function C	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

【図 11】

①一時停止前

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
2	Function B	制御	実行中
3	Function C	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

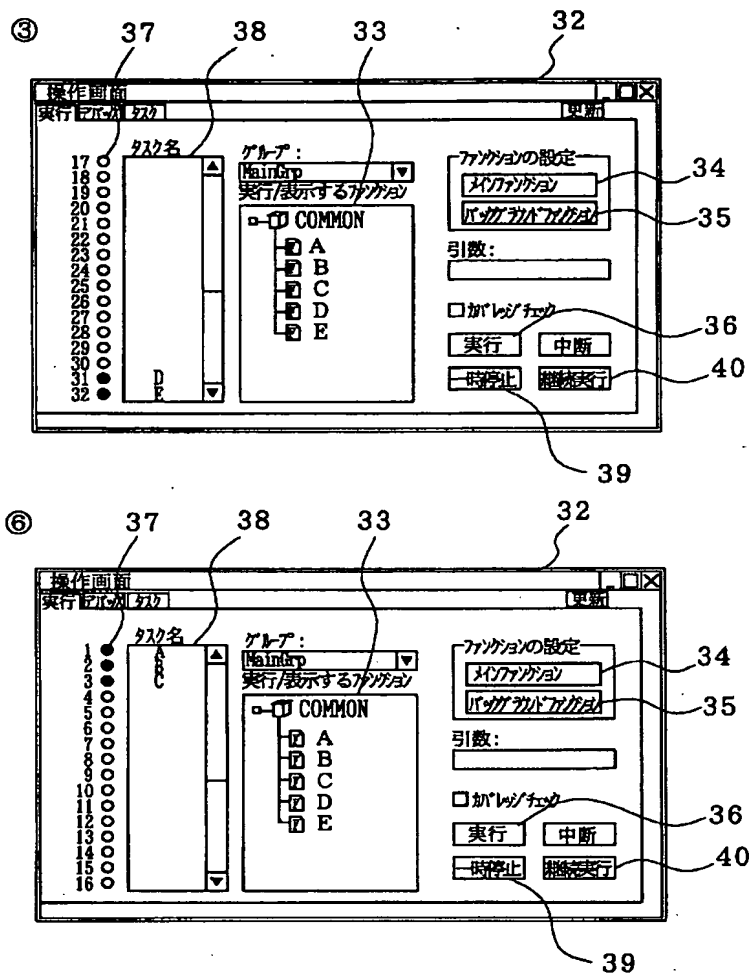
②一時停止中

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	一時停止
2	Function B	制御	一時停止
3	Function C	制御	一時停止
31	Function D	監視	実行中
32	Function E	監視	実行中

③再開後

タスク番号	ファンクション名	グループ	実行状態
1	Function A	制御	実行中
2	Function B	制御	実行中
3	Function C	制御	実行中
31	Function D	監視	実行中
32	Function E	監視	実行中

【図7】



【図10】

